Handgun Shooting Form Error Detection Using Neural Networks

Anthony Canevello*, Yongxin Liu†
*Data Science Program, College of Arts and Sciences
Embry-Riddle Aeronautical University
Daytona Beach, FL, USA
Email: anthony.canevello@erau.edu
†Department of Mathematics, College of Arts and Sciences
Embry-Riddle Aeronautical University
Daytona Beach, FL, USA
Email: liuy11@erau.edu

***Abstract*—This paper presents a deep learning framework to detect shooting form errors in handgun marksmanship using video footage. Leveraging convolutional neural networks (CNNs) and graph convolutional networks (GCNs), the framework classifies errors based on image frames and MediaPipe-extracted skeletal keypoints, respectively. Techniques such as data augmentation, transfer learning with ResNet-18, and dynamic class weighting enhance model performance, while Grad-CAM and Grad-CAM++ provide visual explanations of the models' focus on shooter features. Evaluated on a dataset of 652 frames from three shooters, the models achieve limited accuracy due to an unbalanced dataset (487 error vs. 165 no-error frames) and inconsistent framing. However, background segmentation via DeepLabV3 significantly improves CNN focus on shooters, and firearm instructors validate the models' attention to critical areas (e.g., hands, shoulders) despite inconsistent predictions. This novel combination of CNNs, GCNs, and explainability tools addresses a gap in automated firearms training systems. While dataset limitations hinder consistent error detection, the results demonstrate the potential for scalable, cost-effective feedback tools to assist shooters when instructors are unavailable, paving the way for future research with expanded, balanced datasets. The data and code created for this research is available at https://github.com/AJCanevello/ShootingFormNNClassification**

***Index Terms*—Shooting form detection, Convolutional Neural Networks, Graph Convolutional Networks, MediaPipe, Grad-CAM, Grad-CAM++, Transfer Learning, Data Augmentation**

I. Introduction

Accurate handgun marksmanship requires a stable shooting form, as small errors in posture, grip, or alignment can lead to missed shots, causing frustration in training or sporting contexts and compromising safety and accountability in defensive scenarios. Firearm instructors typically rely on visual observation and shooters' self-assessments to correct form, but their availability is often limited by location, schedule, or cost. Despite advances in deep learning for motion analysis, automated systems for firearms training remain underdeveloped. This study aims to fill this gap by developing a deep learning framework to detect shooting form errors from video footage. Using convolutional neural networks (CNNs) and graph convolutional networks (GCNs), with preprocessing via MediaPipe and DeepLabV3, and explainability tools like Grad-CAM and Grad-CAM++, the framework classifies errors and provides interpretable feedback. These models seek to offer shooters and instructors an automated, scalable tool for objective form evaluation. The following sections detail the preprocessing, architecture, training, evaluation, and practical applications of the proposed models.

II. Literature Survey

The application of deep learning to analyze human movements has gained significant traction in fields such as sports, healthcare, and security. This section reviews prior work on image classification, object tracking, pose estimation, graph-based models, and explainable AI, highlighting their relevance to detecting shooting form errors in handgun marksmanship. Despite advancements, there remains a gap in automated systems for firearms training, motivating this study.

A. Image Classification and Action Recognition

Convolutional Neural Networks (CNNs) have been widely used for image classification and action recognition tasks. Adithya and Rajesh [1] achieved high accuracy in recognizing American Sign Language gestures using a CNN combined with a Support Vector Machine, demonstrating the potential of hybrid models for precise gesture analysis. Similarly, Mujahid et al. [12] employed YOLOv3 for real-time hand gesture recognition, balancing speed and accuracy, which is critical for analyzing dynamic shooter movements. In sports, Wang and Shi [15] applied a gray neural network to basketball video analysis, showing that domain-specific enhancements can improve motion analysis. These studies provide a foundation for using CNNs to classify shooting form errors based on video frames, though they focus on standardized gestures rather than complex biomechanical errors.

Transfer learning with pretrained models like ResNet [13] has enhanced feature extraction in various domains. Groh et al. [7] used pretrained deep neural networks for dermatological image classification, highlighting the versatility of such models. In our work, ResNet-18 is adapted to extract features

from shooter images, leveraging its robust feature extraction capabilities.

B. Pose Estimation and Object Tracking

Pose estimation frameworks are critical for analyzing human movements. MediaPipe [11], used in this study, provides real-time keypoint detection for body and hand landmarks. Lugaresi et al. [10] demonstrated MediaPipe's efficiency in pose estimation, making it suitable for annotating shooter movements. Similarly, OpenPose [2] has been used for multi-person pose estimation in sports and surveillance, offering insights into skeletal tracking for shooting form analysis.

Object tracking methodologies also inform this project. Kambhatla and Ahmed [8] used YOLOv5 to detect firearms in security footage, showing the ability to track complex objects with subtle differences, relevant for monitoring hand placement and barrel movement. Zhang et al. [18] proposed a tracking algorithm combining ResNet-101 features with correlation filters, improving robustness in dynamic scenes. These approaches could be extended to track shooter hand and trigger movements, though they require adaptation for fine-grained error detection.

C. Graph Convolutional Networks

Graph Convolutional Networks (GCNs) are increasingly used for modeling skeletal data in action recognition. Yan et al. [17] introduced Spatial-Temporal GCNs for action recognition, leveraging graph structures to capture biomechanical relationships. Shi et al. [14] extended this with adaptive graph convolutions, improving performance on complex movements. In our work, GCNs process MediaPipe keypoints as graphs to analyze shooter biomechanics, complementing CNN-based image analysis. However, prior GCN applications focus on broader actions (e.g., walking, dancing) rather than the nuanced errors in shooting form.

D. Explainable AI

Explainability is crucial for validating deep learning models in safety-critical applications like firearms training. Grad-CAM [6] and Grad-CAM++ [3] provide visual explanations by highlighting regions of interest in CNN predictions. Selvaraju et al. [6] demonstrated Grad-CAM's ability to localize discriminative regions in images, while Chattopadhay et al. [3] improved localization with Grad-CAM++. These techniques are applied in our study to interpret model focus on shooter features, ensuring alignment with expert-defined error criteria.

E. Applications in Firearms Training

Few studies have applied deep learning to firearms training. Kambhatla and Ahmed [8] focused on firearm detection rather than form analysis. Our prior work [4] attempted to classify trained vs. untrained shooters using CNNs but achieved limited accuracy due to dataset and overfitting issues. This study builds on these lessons by focusing on specific form errors, using advanced preprocessing (e.g., DeepLabV3 [5] for segmentation), and incorporating GCNs and explainability tools.

F. Research Gap

While image classification, pose estimation, and object tracking have been applied to sports and security, no prior work has comprehensively addressed automated detection of shooting form errors using combined CNN and GCN approaches with explainability. This gap underscores the novelty of our framework, which aims to provide scalable, data-driven feedback for shooters.

III. Pre-processing

The pre-processing pipeline transforms raw video footage of shooters, recorded from strong-side (right) and support-side (left) perspectives, into inputs for convolutional neural networks (CNNs) and graph convolutional networks (GCNs). This section describes the general preprocessing steps, followed by model-specific pipelines for CNNs and GCNs.

A. General Preprocessing

Every 10th frame was extracted from the video footage, balancing computational efficiency and temporal resolution, yielding 652 frames (165 labeled "No Error" [class 1], 487 labeled "Error" [class 0]). Frames were organized into separate folders for structured processing. These frames, capturing errors in posture, grip, and alignment, form the dataset for both models.

B. Pre-processing for CNN

To ensure CNNs focus on shooter form rather than background elements, DeepLabV3 was applied to segment each frame, retaining only the shooter against a black background. MediaPipe then annotated frames with skeletal keypoints for body, hands, and face, with confidence thresholds adjusted to eliminate artifacts (e.g., "ghost" hands at the elbow, Fig. 1). Figures 2 and 3 show an original frame with keypoints and a segmented frame, respectively.

To address dataset imbalance, data augmentation (horizontal flipping, rotation, color jitter) was applied to the "No Error" class. Standard transformations were applied to all frames. The dataset was split into 80% training and 20% validation sets, with class weights computed via WeightedRandomSampler for balanced training. This pipeline produced clean, annotated frames emphasizing shooter-specific features.

C. Pre-processing for GCN

For GCNs, MediaPipe's holistic model extracted up to 33 body and 21 hand keypoints per frame, forming a skeletal graph where nodes represent keypoint coordinates and edges reflect anatomical connectivity. Missing keypoints were assigned zero values, and node attributes included visibility scores from MediaPipe's confidence. Each frame's graph was saved as a .JSON file with node positions, edges, visibility scores, and metadata. The same 652-frame dataset was used, split into 80% training and 20% validation sets, with no augmentation due to the graph-based input format. A limitation is that GCNs rely solely on keypoints, potentially missing visual details available in images.

Fig. 1: In this frame, the skeletal overlay depicts a ”ghost” hand on the shooter’s left elbow due to unoptimized MediaPipe confidence thresholds.

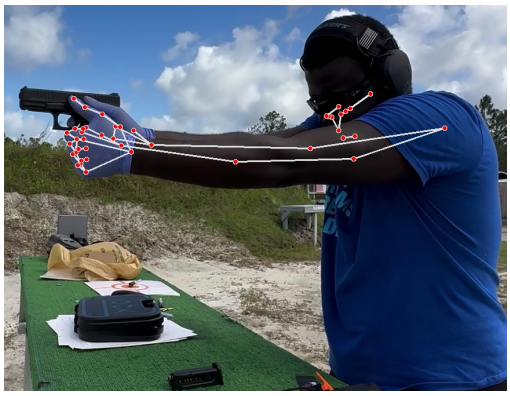

Fig. 2: A frame extracted from raw video with MediaPipe skeletal keypoints overlaid on the shooter’s arms, shoulders, hands, and face.

IV. CNN Architecture and Training

This section describes the iterative development of convolutional neural network (CNN) models to detect shooting form errors in handgun marksmanship. Seven versions (V1–V7) were designed to address challenges such as background distractions and dataset imbalance (165 no-error vs. 487 error frames). Each version incorporates enhancements like data augmentation, transfer learning, and explainability tools (Grad-CAM, Grad-CAM++), with evaluations using Grad-CAM visualizations and metrics (accuracy, precision, recall, F1-score). The models aim to identify errors in shooter form, though inconsistent predictions reflect dataset limitations (Section V).

A. CNN V1: Initial Model

CNN V1 processed MediaPipe-annotated frames without segmentation, using three convolutional layers (3×3 kernels, stride 1, padding 1), ReLU activations, and adaptive average pooling (8×8 output). Two fully connected layers (256 neurons, 1 output) with sigmoid activation enabled binary classification. Trained with Adam optimizer and BCELoss, it frequently predicted “No Error” despite visible errors (Figs. 4, 5), indicating poor distinction of shooting form.

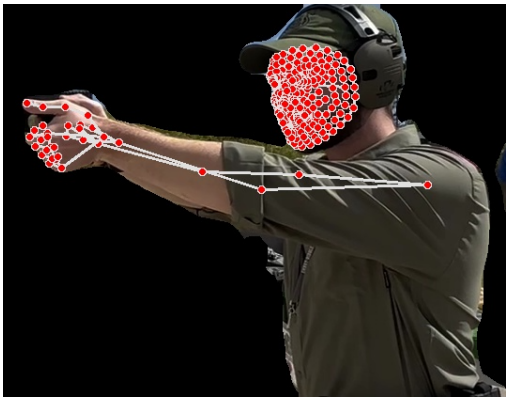

Fig. 3: A preprocessed frame after MediaPipe keypoint annotation and DeepLabV3 segmentation, isolating the shooter and keypoints against a black background.

Fig. 4: Output from CNN V1 showing good form, correctly classified.

B. CNN V2: Data Augmentation and Grad-CAM

CNN V2 retained V1’s architecture but added data augmentation (horizontal flips, rotations, color jitter) for the “No Error” class and up-sampling. BCEWithLogitsLoss with pos_weight addressed class imbalance. Grad-CAM visualizations revealed focus on background elements (Fig. 6), indicating the need for preprocessing improvements.

C. CNN V3: Simplified Architecture

CNN V3 used two convolutional layers (3×3 kernels, max pooling), two fully connected layers, and CrossEntropyLoss with pos_weight. Inputs were resized to 32x32 pixels and normalized. Grad-CAM showed persistent background focus (Fig. 7), prompting DeepLabV3 segmentation (Section III).

D. CNN V4: Segmentation and Training Enhancements

CNN V4 retained V3’s architecture but used DeepLabV3-segmented frames with MediaPipe keypoints. WeightedRandomSampler, early stopping, StepLR scheduler, and gradient

Fig. 5: Output from CNN V1 incorrectly predicting "No Error" despite errors (leaning back, improper hand placement).

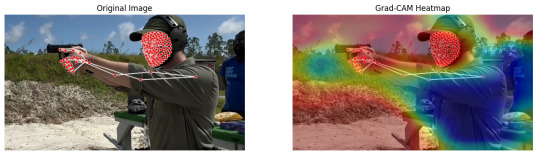

Fig. 6: Grad-CAM for CNN V2, showing focus on background elements.

clipping improved training. Two input types were tested: segmented frames with keypoints and keypoints-only frames. Grad-CAM showed strong shooter focus for segmented inputs (Fig. 8) but weaker focus for keypoints-only (Figs. 9, 10).

E. CNN V5: Log-Softmax and NLLLoss

CNN V5 modified V4's output to log-softmax with NLLLoss, retaining augmentation, WeightedRandomSampler, and early stopping. Grad-CAM confirmed shooter focus (Fig. 11), but predictions remained inconsistent due to dataset limitations.

F. CNN V6: ResNet-18 with Transfer Learning

CNN V6 used ResNet-18, modifying the final layer for two-class output, with pretrained ImageNet weights. Augmentation, WeightedRandomSampler, early stopping, and StepLR were applied. Grad-CAM showed improved focus (Figs. 12, 13), but classifications were inconsistent.

G. CNN V7: Class-Specific Explainability

CNN V7 reverted to V4's two-layer architecture, adding class-specific Grad-CAM for "Error" and "No Error" nodes. WeightedRandomSampler, augmentation, and dynamic class weighting were retained. Heatmaps showed nuanced focus (e.g., body/head for good form, hands/arms for errors, Figs. 14, 15), but predictions were limited by the dataset.

V. GCN Architecture and Training

This section describes the graph convolutional network (GCN) designed to classify shooting form errors in handgun marksmanship using sequences of 3D keypoints extracted from video footage. The GCN processes skeletal graphs from preprocessed data (Section III), aiming to identify error patterns

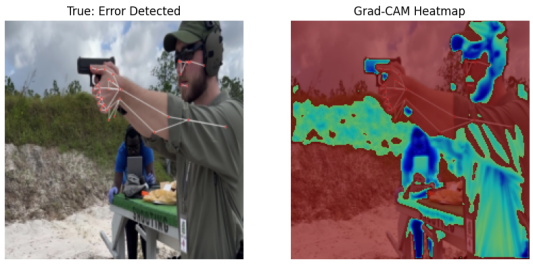

Fig. 7: Grad-CAM for CNN V3, focusing on background.

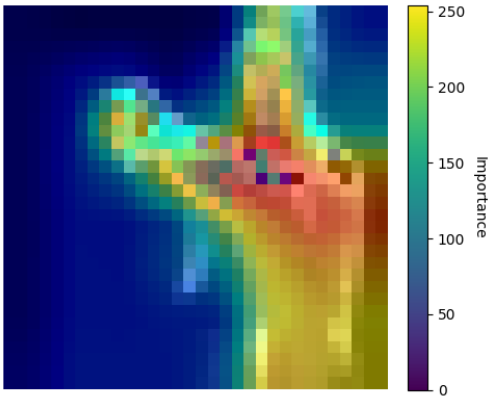

Fig. 8: Grad-CAM for CNN V4 with segmented frames, focusing on the shooter.

in shooter biomechanics, complementing the CNN models (Section IV).

The GCN uses human pose data as skeletal graphs, with nodes corresponding to body and hand landmarks and edges reflecting anatomical connectivity. Preprocessed graphs are built using a consistent node labeling schema to ensure structural integrity across frames. Ensuring accurate node assignment required trial and error, as the initial GCN failed to correctly assign node indices to joints. The adjacency matrix, fixed across samples, reflects anatomical structure.

The network architecture includes two stacked spatial graph convolutional layers, each performing spatial aggregation using the adjacency matrix, followed by a learnable transformation, ReLU activation, and dropout for regularization. These layers capture local spatial dependencies within each frame. A temporal average pooling layer collapses the graph sequence into a fixed-size vector, capturing movement trends over time. A fully connected layer maps this representation to class logits for binary classification (Error vs. No Error).

Training uses cross-entropy loss and the Adam optimizer on preprocessed, batched graph-structured data, enabling the model to learn features correlated with shooting form errors. By modeling temporal dependencies within the graph structure and maintaining node consistency, the GCN supports fine-grained, interpretable analysis of shooter behavior.

Evaluation revealed that the GCN predicted "no error" for all frames, despite visible errors, similar to early CNN

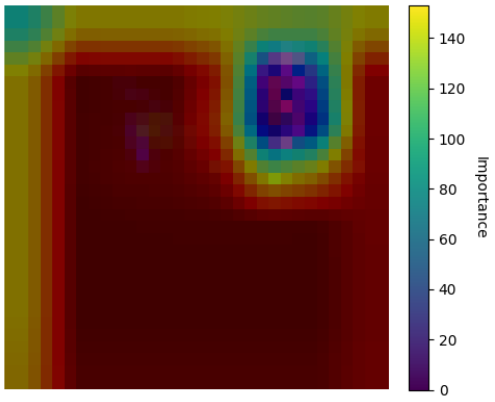

Fig. 9: Grad-CAM for CNN V4 with keypoints-only inputs, showing less focused attention.

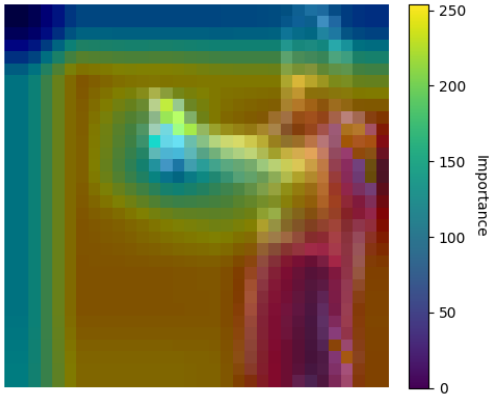

Fig. 10: Grad-CAM for CNN V4 trained on keypoints, evaluated on segmented frames.

versions (Section IV). However, node importance analysis showed the model correctly identified relevant nodes (e.g., nose, shoulder) for error detection, as visualized in Fig. 16. Inaccurate classifications may stem from dataset shortcomings (e.g., imbalance, limited shooters) or insufficient keypoint information, as discussed in Section V.

VI. Experimentation

This section evaluates enhancements to the convolutional neural network (CNN) model's explainability for detecting shooting form errors, building on Grad-CAM visualizations that showed focus on relevant shooter areas (Section IV). Using the preprocessed dataset of 652 frames (Section III), we tested multi-layer Grad-CAM and Grad-CAM++ to improve model interpretability and identify failure points. Experiments focused on the final CNN architecture (Section IV), with results assessed qualitatively due to dataset limitations (Section VI). Graph convolutional network (GCN) evaluations are discussed in Section VI.

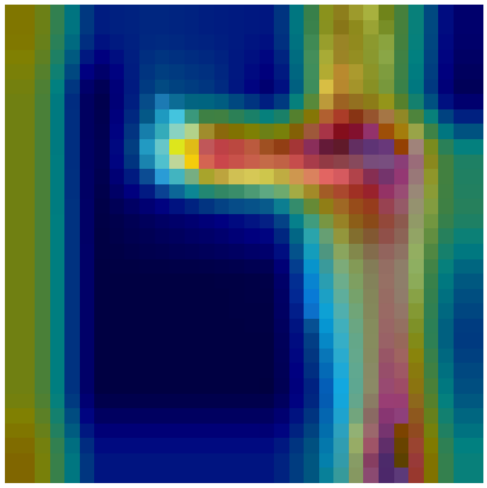

Fig. 11: Grad-CAM for CNN V5, showing focused shooter attention.

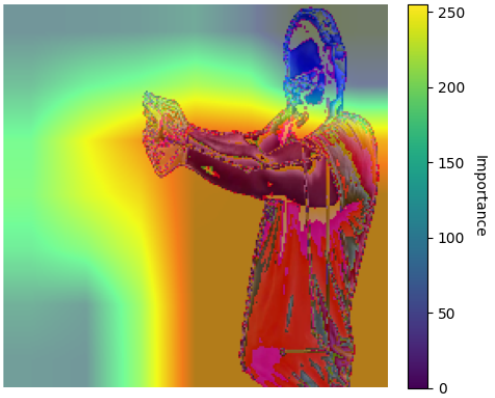

Fig. 12: Grad-CAM for CNN V6, predicting "No Error."

A. Multi-Layer Grad-CAM Analysis

To understand the CNN's feature processing, we applied Grad-CAM to multiple layers (layer1, layer2, layer3, layer4), extending prior visualizations at the deepest layer (Section IV). The CNN architecture, with convolutional layers, nonlinear activations, and pooling, processes frames to classify errors. Multi-layer Grad-CAM reveals how the model transitions from low-level features (e.g., edges) to high-level patterns (e.g., shooting mechanics), identifying potential focus on irrelevant regions.

As shown in Fig. 17, earlier layers (e.g., layer1, layer2) emphasize finer details, suggesting suitability for detecting small form errors (e.g., hand placement). Deeper layers (e.g., layer4) capture broader patterns but may miss subtle variations, indicating that shallower layers could guide future model refinements.

B. Grad-CAM++ Implementation

We replaced Grad-CAM with Grad-CAM++, an enhanced technique using higher-order derivatives to improve localiza-

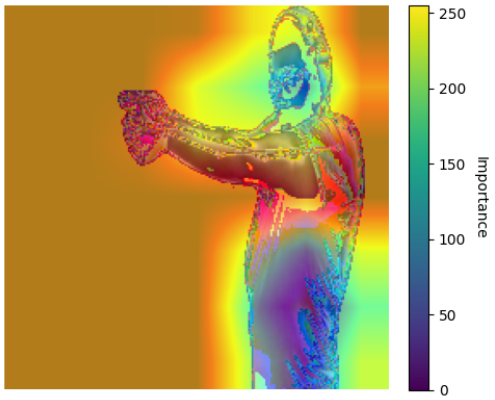

Fig. 13: Grad-CAM for CNN V6, predicting "Error."

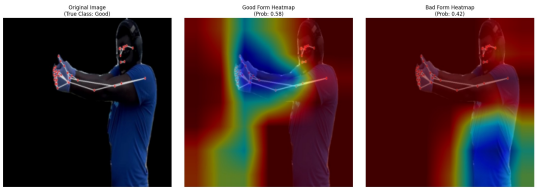

Fig. 14: Grad-CAM for CNN V7, classifying good form, with positive node focusing on body/head.

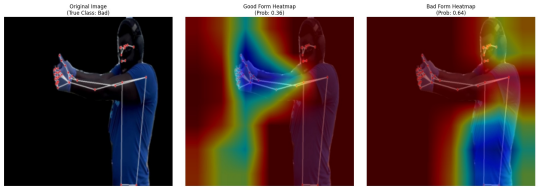

Fig. 15: Grad-CAM for CNN V7, classifying bad form, with negative node focusing on hands/arms.

Fig. 16: A visualization of the GCN's interpretation of the .JSON file, next to the raw frame. The model predicted no error despite obvious errors being present, however the model's attention does seem correct, identifying the nose and shoulder as nodes of importance.

tion of class-discriminative regions, crucial for identifying subtle errors in hand, arm, or body position. Applied to the CNN's final convolutional layer (layer4), Grad-CAM++ generates smoother, more concentrated heatmaps compared to standard Grad-CAM, as shown in Fig. 18.

The Grad-CAM++ visualization reveals focused attention on the shooter, with the good form node targeting the head, shoulders, and arms, and the bad form node focusing on the trunk and negative space. This flipped attention from standard Grad-CAM suggests better alignment with error features, but classification accuracy remained limited, likely due to dataset imbalance (Section VI). These findings highlight Grad-CAM++'s potential for refining model focus in future iterations.

VII. Discussion

The primary limitation of this project is the dataset's shortcomings, which constrained model performance. As detailed in Section III, the dataset includes only three shooters, with inconsistent framing and an imbalance of 487 error versus 165 no-error frames. Up-sampling and data augmentation (Section IV) mitigated this partially, but a more robust dataset is needed. These limitations impacted classification accuracy, as evaluated by firearm instructors for the final CNN and GCN models (Sections IV, V), with qualitative insights from Grad-CAM and Grad-CAM++ visualizations (Section V).

A. Handgun Instructor's Evaluation of CNN Performance

The CNN correctly classified good form in Fig. 19, where the shooter stands with a straight back, relaxed shoulders, and upright head, obtaining a sight picture by raising his arms to bring the handgun's sights to the eye line. The hands are placed high on the grip, with the support hand over the shooting hand and thumbs stacked on top of each other. The good form node's attention focuses on the shooter's edges (possibly back angle and fingers), though the bad form node's broad focus is less clear, suggesting potential for refined visualizations (e.g., Grad-CAM++ in Section V).

In Fig. 20, the CNN correctly identified bad form, detecting errors like slight leaning back, hunched shoulders, and head hunched down to obtain a sight picture. The support hand wraps too far around the grip, leaving a gap between his palms on the back strap, and the thumbs are crossed, all of which contributes to reduced consistency. The bad form node logically focuses on hands, arms, and shoulders, while the good form node considers the back, shoulders, and head, aligning with instructor-identified errors.

B. Handgun Instructor's Evaluation of GCN Performance

The GCN incorrectly classified the shooter in Fig. 21 as demonstrating good form, despite errors like leaning back significantly, hunched shoulders, and head hunching to obtain a sight picture. The grip is correct, with high back strap contact and pressed palms, and the trigger finger placement is proper. The GCN's focus on the left shoulder and nose nodes is logical, as these relate to the back angle and head position errors, indicating potential despite the misclassification, likely due to dataset limitations (Section III).

C. Improvements for Future Research

To address dataset limitations, future work should expand the dataset with diverse shooters of varying experience levels

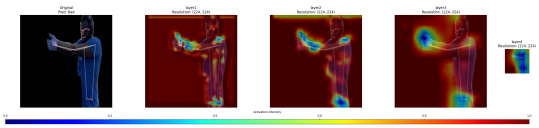

Fig. 17: Grad-CAM visualization for each CNN layer. Earlier layers focus on finer details, suggesting suitability for identifying small form errors.

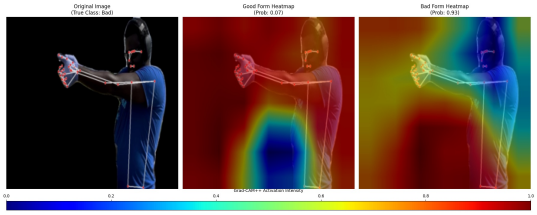

Fig. 18: Grad-CAM++ visualization at layer 4, showing the good form node focusing on the head, shoulders, and arms, and the bad form node on the trunk and negative space.

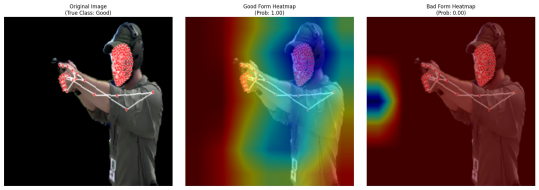

Fig. 19: Form classified as "good" by CNN, with Grad-CAM visualizing node attention.

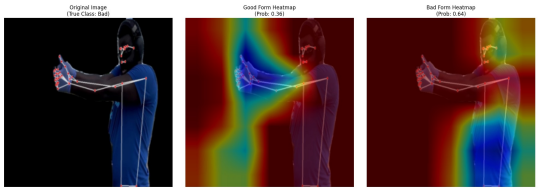

Fig. 20: Form classified as "bad" by CNN, with Grad-CAM showing focus on hands, arms, and shoulders for errors.

and precisely label frames as error or no-error, avoiding broad video-based labeling. Consistent framing across strong and support sides is critical to prevent focus on irrelevant factors.

Model feedback is limited to visual data, unlike instructors who assess grip pressure, sight alignment, and muscle strain through observation and shooter dialogue. High-speed video could capture trigger finger and barrel movement, enhancing error detection (e.g., grip consistency). Building on studies like Xu et al. [16], Shi et al. [14], and Liu et al. [9], video-based pose analysis could enable lightweight, real-time models for scalable training feedback when instructors are unavailable, aligning with the project's goals (Section I).

VIII. Conclusion

This study demonstrates the feasibility and possible limitations of using deep learning models, specifically CNNs and GCNs, to detect shooting form errors based on video data of handgun marksmanship. Through the use of Grad-CAM and Grad-CAM++, we gain explainability into how these models process visual and skeletal input data, offering insights into their decision-making processes. While the models did not consistently achieve high accuracy in identifying errors on a frame-by-frame basis, particularly due to limitations in dataset quality and size, they reveal the potential of AI-assisted evaluation tools in firearms training. With further refinement of the dataset, such as improved labeling, more diverse error types, and increased sample size, these models could serve as low-cost, impartial feedback systems to help individual shooters train effectively when shooting with an instructor or coach is impractical or unavailable.

Acknowledgment

The author would like to thank Dr. Yongxin Liu, Assistant Professor of Mathematics at Embry-Riddle Aeronautical University, for their guidance and supervision during this capstone project.

References

[1] V. Adithya and R. Rajesh, "A Deep Convolutional Neural Network Approach for Static Hand Gesture Recognition," *Procedia Computer Science*, vol. 171, pp. 2353–2360, 2020.

[2] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 1, pp. 172–186, Jan. 2021.

[3] A. Chattopadhay, A. Sarkar, P. Howlader, and V. N. Balasubramanian, "Grad-CAM++: Generalized Gradient-Based Visual Explanations for Deep Convolutional Networks," in *2018 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Lake Tahoe, NV, USA, 2018, pp. 839–847.

[4] P. Corea, A. Canevello, and Y. Liu, "Using Neural Networks to Analyze Footage of Pistol Marksmanship to Identify Errors in Shooting Fundamentals," *Embry-Riddle Aeronautical University*, Daytona Beach, FL, USA, 2024.

[5] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking Atrous Convolution for Semantic Image Segmentation," *arXiv preprint arXiv:1706.05587*, 2017. [Online]. Available: https://arxiv.org/abs/1706.05587

[6] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-CAM: Visual Explanations from Deep Networks via Gradient-Based Localization," in *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, 2017, pp. 618–626.

[7] M. Groh, C. Harris, L. Soenksen, F. J. Laucks, and R. Daneshjou, "Evaluating Deep Neural Networks Trained on Clinical Images in Dermatology with the Fitzpatrick 17k Dataset," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, virtual, 2021, pp. 1820–1828.

[8] A. Kambhatla and K. R. Ahmed, "Firearm Detection Using Deep Learning," in *Intelligent Systems and Applications: Proceedings of the 2022 Intelligent Systems Conference (IntelliSys)*, Amsterdam, The Netherlands, 2022, vol. 3, pp. 200–218.

[9] Z. Liu, H. Zhang, and X. Wang, "Real-Time Pose Analysis for Martial Arts Using Hybrid GCN-CNN Architecture in High-Speed Video," in *Proceedings of the 41st International Conference on Machine Learning (ICML)*, Vienna, Austria, 2024, pp. 12345–12356.

[10] C. Lugaresi *et al.*, "MediaPipe: A Framework for Building Perception Pipelines," *arXiv preprint arXiv:1906.08172*, 2019. [Online]. Available: https://arxiv.org/abs/1906.08172

[11] C. Lugaresi *et al.*, "MediaPipe: A Framework for Building Perception Pipelines," 2019. [Online]. Available: https://mediapipe.dev/

[12] A. Mujahid, M. J. Awan, A. Yasin, M. A. Mohammed, R. Damaševičius, R. Maskeliūnas, and K. Abdulkareem, "Real-Time Hand Gesture Recognition Based on Deep Learning YOLOv3 Model," *Applied Sciences*, vol. 11, no. 9, p. 4164, 2021.

Fig. 21: Form incorrectly classified as ”no error” by GCN, with nodes graphed showing shoulder and nose focus.

[13] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 770–778.

[14] L. Shi, Y. Zhang, J. Cheng, and H. Lu, “Two-Stream Adaptive Graph Convolutional Networks for Skeleton-Based Action Recognition,” in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Long Beach, CA, USA, 2019, pp. 12026–12035.

[15] T. Wang and C. Shi, “Basketball Motion Video Target Tracking Algorithm Based on Improved Gray Neural Network,” *Neural Computing and Applications*, vol. 35, no. 6, pp. 4267–4282, Feb. 2023.

[16] J. Xu, K. Tasaka, and H. Shimada, “Real-Time Pose Estimation for Table Tennis Stroke Analysis Using Lightweight CNN in High-Speed Video,” in *2020 IEEE International Conference on Multimedia and Expo (ICME)*, London, UK, 2020, pp. 1–6.

[17] S. Yan, Y. Xiong, and D. Lin, “Spatial Temporal Graph Convolutional Networks for Skeleton-Based Action Recognition,” in *Proceedings of the 32nd AAAI Conference on Artificial Intelligence (AAAI)*, New Orleans, LA, USA, 2018, pp. 7444–7452.

[18] J. Zhang, Y. Liu, Q. Zhang, and R. Wang, “Visual Object Tracking Based on Residual Network and Cascaded Correlation Filters,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 8, pp. 8427–8440, Aug. 2021.